

**BEACON HR/PAYROLL IMPLEMENTATION PROJECT  
ENHANCEMENTS STRATEGY**

## TABLE OF CONTENTS

<b>Document History .....</b>	<b>Error! Bookmark not defined.</b>
<b>Approvers .....</b>	<b>Error! Bookmark not defined.</b>
<b>1. Enhancements and Modifications .....</b>	<b>1</b>
1.1. Functional configuration .....	2
1.2. Development object creation .....	2
<b>2. Strategy .....</b>	<b>3</b>

## 1. ENHANCEMENTS AND MODIFICATIONS

Enhancements are a natural by-product of the development process. SAP's strength is its flexibility in adapting to individual customer requirements without having to change the core application.

User-developed customizations, such as configuration changes and code inserted at pre-defined points, are maintained separately from the core applications, and if properly applied can easily be preserved through an upgrade. Only changes to the core applications (that is, Modifications to the source code) are not automatically upgraded. However, even in this case, an audit is performed before upgrading the core applications, and any changes are highlighted to allow the changes to be extracted and reinstated.

SAP provides three mechanisms to modify its software:

- **Configuration.** SAP provides a table-driven approach to the selection of options already embedded into the product. This type of change does not require programming skills.
- **Enhancements.** SAP provides standard entry points or user exits within the application where customers can, for example, make calls to external programs, gather data, and do analysis. User exits allow a further degree of flexibility in how the delivered program code operates. Customers may also write their own program code based upon the architecture designed into the user exit. Additional logic can also be added by customer written programs that utilize the SAP programming environment and are managed independently from SAP provided objects. Naming conventions will be in place to isolate customer objects from vendor supplied objects.
- **Modification.** A modification is the act of changing or modifying the actual source code. SAP does not encourage the modification of delivered program code because this results in future maintenance, support issues for the customer, and increase maintenance costs.

Change management and upgrade tools are provided with the software. The system automatically tracks any configuration settings or enhancements made. The provided upgrade tools automatically compare a customer's changes with the delivered upgrade in order to identify any potential conflicts. Configuration settings and enhancements can be automatically migrated to upgraded versions of the software. However, modifications (changes to the actual source code) are not automatically migrated to upgraded versions of the software. Modifications that customers want to keep and whose corresponding objects in the new release have also been changed by SAP must be merged with the new SAP object. This merging is known as modification adjustment. Modifications are tracked separately from enhancements in the change management tool, named the Transport Management System. Enhancements are tracked as changes in Transports, while modifications are tracked as 'Repairs'. All Repairs have to be registered with SAP

|

before implementing them. This aids SAP in providing support by keeping them aware of customer applied modifications to core code, while they are responding to a problem that has been logged.

### **1.1. Functional Configuration**

To have the standard system meet a customer's needs, SAP software need only be "configured," as opposed to "modified." The mySAP solutions are highly customizable, rules-based applications. Elements such as screens, logic, rules, processes, drop-down lists, and data values can be customized through delivered integrated customization tools. Configurations (business process changes, and rule definitions) are handled through the IMG.

The SAP IMG is a template-driven approach toward defining rules. Configuration of rules consists of answering online questions and filling in online forms. Coding is not required. Using the SAP IMG, subject matter experts on the business side of the organization take responsibility for directly setting up business rules.

### **1.2. Development Object Creation**

For system enhancement and modification, the SAP ABAP/4 Objects Development Workbench is SAP R/3's development environment for creation of enterprise-wide client/server business solutions. It supports the entire software development cycle with tools for modeling, programming in the 4GL language ABAP Objects, definition of data and table structures, and the design of GUIs. Extensive resources for software testing, tuning, and maintenance, as well as for the work of large development teams, are also available. As a supplement to the development tools, SAP delivers a library of pre-built business and software components that can easily be incorporated into custom programs. Version control tools are integrated with the Workbench.

SAP defines a user exit as a point in an SAP program where a customer's own program can be called. In contrast to customer exits, user exits allow developers to access and modify program components and data objects in the standard SAP system. On upgrade, each user exit must be checked to ensure that it conforms to the standard system.

There are the following types of user exit:

I

- **User exits that use includes.** These are customer enhancements that are called directly in the program
- **User exits that use tables.** These are used and managed with Customizing.

The SAP system includes an integrated Data Dictionary tool. The Data Dictionary contains both technical and business definitions of all tables, views, indices, fields, and data elements delivered with the system. The Data Dictionary also allows customers to extend the data model with additional tables, fields, views, indices, and data elements. All actions necessary to modify the data model are taken via the Data Dictionary. Direct access to low-level database tools is not required. The Data Dictionary tracks all vendor supplied code as well as any customer developed objects. The end user can not distinguish between a vendor supplied object and a customer developed object when using the system.

## 2. STRATEGY

The overall strategy for the Beacon project will be to utilize configuration and SAP provided exit points to add NC specific logic. If additional requirements are identified that require custom objects to be created, then the functional team will work with the end users to define the specific logic / functions that need to be added and the development team will develop the objects according to project programming and naming standards.

Modifications require PMO approval with documentation and justification being provided by the requesting sub team, giving the reason the modification is needed. If approved by the PMO, then the modification will occur following programming and naming standards, and will be registered with SAP.

I